

SHOWCB Enhancements in z/VSE 5.1

This edition applies to Version 9 Release 1 of IBM VSE/VSAM, which is part of VSE Central Functions, Program Number 5686-CF9, and to all subsequent releases and modifications unless otherwise indicated in new editions.

© **Copyright IBM Corporation 2012.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Table of Contents

Trademarks.....	3
Overview.....	4
Format of the SHOWCB Macro.....	4
LSR Matrix.....	5
Header.....	5
Share Pool Statistics Area.....	6
String Statistics Area.....	6
Buffer Matrix.....	7
Cluster Matrix.....	8
Usage.....	9
Extent Matrix.....	9
Header.....	9
Physical Device Characteristics Area.....	10
Extent Information Area.....	11
Usage.....	12
Example of an LSR Matrix Call.....	13
LSR Input Information	
.....	13
LSR Matrix Output.....	14
Example of an Extent Matrix Call.....	15
Extent Matrix Output.....	15
New Fields Supported by the SHOWCB Macro.....	16
Example of a SHOWCB Call.....	16
.....	17
Remarks.....	17
Comments and Questions.....	17

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies.

A current list of IBM trademarks is available on the web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Overview

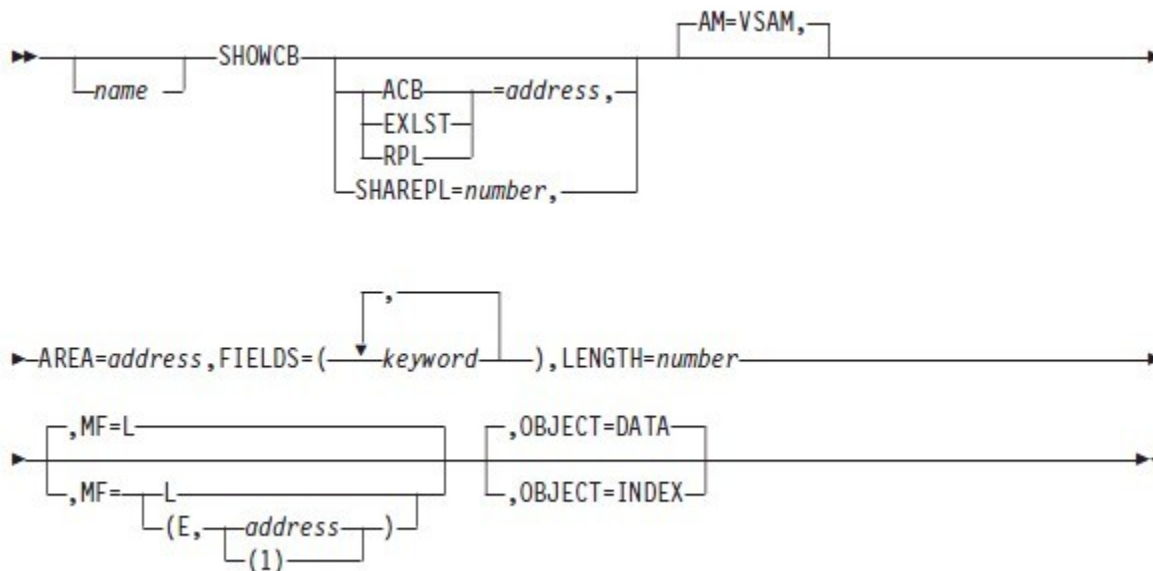
The SHOWCB macro displays fields in an ACB, EXLST, or RPL control block. SHOWCB places these fields in an area that you provide. They are independent of the format of the control block or list you are displaying. The fields are displayed in the order that you specify the keywords for them.

All attributes supported by the SHOWCB macro up to z/VSE 4.3 are described in the “*SHOWCB Macro Section*” of Chapter 12 “*Descriptions of VSE/VSAM Macros*” in the “*VSE/VSAM User’s Guide and Application Programming*”.

Starting with z/VSE V5.1, the SHOWCB macro also provides the following information:

- LSR (Local Shared Resources) matrix which contains string statistics information, information about each buffer pool defined for the specified LSR pool, and LSR string and buffer statistics for each cluster within a specified share pool.
- Extent matrix which contains characteristics of physical devices on which the specified cluster resides and information about all extents for the specified cluster.
- Nine new keywords for the FIELDS parameter.

Format of the SHOWCB Macro



AREA=address specifies the address of the area in virtual storage that you are providing for VSE/VSAM to display the items you specify in the FIELDS operand. The items are in the area in the order you specify the keywords. The area must begin on a fullword boundary.

LENGTH=number specifies the length of the display area you are providing.

ACB | EXLST | RPL=address | SHAREPL=number - this operand specifies whether you want to

display an ACB, an EXLST, an RPL, or information about an LSR share pool. In the standard and list forms of SHOWCB, you can omit this operand if you are displaying only the standard length of a control block or list. With the execute form of SHOWCB, you can change the address of the block or list to be displayed, but not the type.

SHAREPL=number specifies the identification number of a Local Shared Resources (LSR) pool to be displayed. Specify a number from 0 through 15.

FIELDS=(keywords) - there are **four** groups of keywords you can code for the FIELDS operand of the SHOWCB:

- The keywords that you can code with the ACB, EXLST, RPL, and GENCB macros.
- The length of ACB, RPL, or EXLST.
- The attributes of an open file or index indicated by ACB.
- The matrix of LSR statistics LSRINF. To retrieve the matrix of LSR statistics into the user's area, LSRINF must be the only keyword in the FIELDS parameter. If SHAREPL=number is specified, OBJECT parameter is ignored.

MF= specifies the list, execute, and generate forms of the control block manipulation macros.

Example of an LSR Matrix call:

```
SHOWCB AREA=USER_AREA, LENGTH=100, SHAREPL=6, FIELDS=(LSRINF)
```

Example of an Extent Matrix call:

```
SHOWCB AREA=USER_AREA, LENGTH=100, ACB=ACb1, FIELDS=(EXTINF)
```

LSR Matrix

Returned LSR matrix consists of three parts:

1. Header.
2. Share Pool Statistics Area: string statistics area which contains total number of LSR strings and buffer matrix which contains buffer statistics for the requested share pool.
3. Cluster Matrix: contains LSR string and buffer statistics for each VSAM cluster assigned to a specified share pool.

Header

Header has a fixed size of 32 bytes and contains the following fields:

Field	Length
Length of area supplied by user	4 bytes

Total length used (or required) by VSAM	4 bytes
Length of string statistics area	4 bytes
Number of rows in buffer matrix	4 bytes
Length of rows in buffer matrix	2 bytes
Number of rows in cluster matrix	4 bytes
Length of rows in cluster matrix	2 bytes
(reserved)	4 bytes
(reserved)	4 bytes

Length of area supplied by user contains the length of the area passed by user in the LENGTH parameter of the MACRO call in bytes.

Total length used (or required) by VSAM contains the length of the area actually used by or needed for VSAM to display string statistics, buffer matrix, and cluster matrix, including length of the header, in bytes.

Length of string statistics area contains the length of fixed string statistics area (first part of share pool statistics area) in bytes.

Number of rows in buffer matrix contains the number of fixed size rows that are displayed in the user's area. This number also indicates the number of subpools in a specified share pool.

Len of rows in buffer matrix contains the length of each row in buffer matrix in bytes.

Number of rows in cluster matrix contains the number of fixed size rows that are passed to the user. This number also indicates the number of clusters in a specified share pool, including base clusters opened via a path.

Length of rows in cluster matrix contains the length of each row in cluster matrix in bytes.

Share Pool Statistics Area

String Statistics Area

This area contains the following information:

Field	Length
Share pool number	2 bytes
Total number of strings	2 bytes
Number of active strings	2 bytes
Number of free strings	2 bytes
(reserved)	2 bytes

(reserved)	2 bytes
(reserved)	2 bytes
(reserved)	2 bytes

Buffer Matrix

Following string statistics area is a series of thirty two byte rows that contain descriptors for each buffer pool defined for this share pool. The fields in a row are the following:

Field	Length
Size of buffer	2 bytes
Type of buffer	1 byte
Flags	1 byte
Number of buffers	4 bytes
Number of modified buffers	4 bytes
Number of free buffers	4 bytes
Number of buffer reads	4 bytes
Number of retry requests without I/O	4 bytes
Number of user-initiated writes	4 bytes
Number of non-user-initiated writes	4 bytes

Size of buffer indicates the size of every buffer in the resource pool. See the “*BLDVRP Macro*” section of Chapter 12 “*VSE/VSAM Macro Descriptions*” in the “*VSE/VSAM User's Guide and Application Programming*” for how to define size of buffers, type of buffers, and number of buffers in the resource pool.

Type of buffer – 'D' means data, 'I' means index.

Flags – reserved field.

Number of buffer reads indicates the number of requests for retrieval that require I/O operation; that is, the data was not found in the buffer. (=KW BUFRDS)

Number of retry-requests without I/O indicates the number of requests for retrieval that could be addressed without an I/O operation; that is, the data was found in the buffer. (=KW BFRFND)

Number of non-user-initiated writes from buffer pool indicates the number of write requests that VSE/VSAM was forced to do because buffers were not available for reading the contents of a control interval (CI). (Number of non-user-initiated writes is the number of write requests that were *not* initiated by the user.) Refer to a description of the NUIW attribute of an open file of the SHOWCB macro in the “*VSE/VSAM User's Guide and Application Programming*”, Chapter 12 “*VSE/VSAM*”

Macros Descriptions".

Number of user-initiated writes indicates the number of all other write requests (those that are not counted in non-user-initiated write requests). Refer to a description of the UIW attribute of an open file of the SHOWCB macro in the "*VSE/VSAM User's Guide and Application Programming*", Chapter 12 "*VSE/VSAM Macros Descriptions*".

Cluster Matrix

LSR string and buffer statistics for each cluster within a specified share pool.

This part contains fixed size rows, number of which equals number of clusters associated with a specified share pool. The length of a row and the current number of rows are contained in the header.

Field	Length
DDNAME	8 bytes
Type of cluster	1 byte
Flags	1 byte
Number of active strings for this cluster	2 bytes
Size of data buffers	4 bytes
Number of data buffers used	4 bytes
Size of index buffers	4 bytes
Number of index buffers used	4 bytes
(reserved)	4 bytes
(reserved)	4 bytes

DDNAME is the name of cluster in a specified share pool. Specifies a character string of up to seven bytes and is the same as the file name parameter specified in the DLBL statement that identifies the VSE/VSAM file or path to be processed.

Type of cluster - 'B' means a base cluster, '00' means path.

Number of active strings for this cluster is the number of active strings for a cluster with the name DDNAME.

Size of data buffers contains the size of data buffers in the resource pool.

Number of data buffers used contains the number of data buffers in the resource pool specified via BLDVRP Macro.

Size of index buffers contains the size of index buffers in the resource pool.

Number of index buffers used contains the number of index buffers in the resource pool specified via BLDVRP Macro.

Usage

The header is 32 bytes length. If the value specified for LENGTH is large enough, VSE/VSAM returns:

- header;
- string statistics area;
- all rows of buffer matrix (their number can be checked in the “Number of rows in buffer matrix” field of the header);
- all rows of cluster matrix (their number can be checked in the “Number of rows of cluster matrix” field of the header).

If LENGTH is not large enough for the whole output (but more than 32 bytes), VSE/VSAM returns as much information as fits, the return code 4, and the reason code 9. Check the field “Total length used (or required) by VSAM” in the header for the required space length, and issue another SHOWCB call specifying LENGTH large enough to contain all the matrix information.

In case user specifies LENGTH less than 32 bytes, VSE/VSAM will reject the request and issue the return code 4, and the reason code 21. Recompile the program with LENGTH bigger than 32 bytes.

Extent Matrix

The output matrix consists of the three parts:

1. Header.
2. Physical Device Characteristics Area which contains information about the extents allocated in the requested ACB.
Note: VSAM requires that all extents for a specific cluster component reside on the same type of DASD. For KSDS and VRDS clusters, the data and index can reside on different types of DASD, so there will be two sets of Physical Device Characteristics, one set used for data and the other used for index.
3. Extent Information Area which contains information about each extent for the requested VSAM cluster. Data extents will be listed first, marked with 'D', followed by the index extents, marked with 'I'.

Header

Header has a fixed size of 32 bytes and contains the following fields:

Field	Length
Length of area supplied by user	4 bytes

Total length used (or required) by VSAM	4 bytes
Length of physical device characteristics area	4 bytes
Number of data extents	4 bytes
Length of data extents row	2 bytes
Number of index extents	4 bytes
Length of data extents row	2 bytes
(reserved)	4 bytes
(reserved)	4 bytes

Length of area supplied by user contains the length of the area passed by user in the LENGTH parameter of MACRO call in bytes.

Total length used (or required) by VSAM contains the length of the area actually used by or needed for VSAM to display physical device information and extent information, including length of the header, in bytes.

Length of physical device information area contains the length of this area in bytes.

Number of data extents contains the number of data extents for the specified cluster (ACB). It also indicates the number of fixed size rows that are displayed in the user's area.

Length of data extents row contains the length of each row in the extent area in bytes.

Number of index extents contains the number of index extents for the specified cluster (ACB). This number also indicates the number of fixed size rows that are displayed in the user's area.

Length of index extents row contains the length of each row in the extent area in bytes.

Physical Device Characteristics Area

This part contains the physical device characteristics for the indicated cluster. Data volume information is displayed first and is followed by index, if applicable. Each 48 bytes contain the following fields:

Field	Length
Volume ID	6 bytes
Type of extent	1 byte
Flags	1 byte
Physical block size	4 bytes
Number of bytes per track	4 bytes
Number of bytes per control area	4 bytes
Number of physical blocks per control interval	4 bytes
Number of physical blocks per track	4 bytes

Number of tracks per control area	4 bytes
Number of tracks per cylinder	4 bytes
Number of physical blocks per control area	4 bytes
(reserved)	4 bytes
(reserved)	4 bytes

- Volume ID – identification of the volume where extents of the current cluster reside.
- **Type of extent** – 'D' if data, 'I' if Index.
- **Flags** – reserved.
- **Number of bytes per track:**
For ECKD this number actually shows number of bytes per track.
For FBA:
 - first 2 bytes contain the number of replications of the control interval that fit on a track.
 - last 2 bytes contain number of the total 'logical' blocks per control area.
- **Number of tracks per control area:**
For ECKD this number actually shows number of tracks per control area.
For FBA this field shows the total number of physical blocks per control area.
- **Number of tracks per cylinder for ECKD:**
For FBA this field is undefined.

Extent Information Area

This part shows information about all extents for a specified file. This part consists of fixed size rows, number of which equals number of extents associated with a specified cluster. The length of each row and the current number of rows can be found in the header. (The length of each row is calculated as number of data extents + number of index extents).

Field	Length
Volser	6 bytes
Type of extent	1 byte
Flags	1 byte
Low extent (CCHH)	4 bytes
(reserved)	4 bytes
High extent (CCHH)	4 bytes
(reserved)	4 bytes
Low RBA	8 bytes
High RBA	8 bytes

(reserved)	4 bytes
(reserved)	4 bytes

Volser is the volume serial number of volume on which extent resides. This is a label assigned when a volume is prepared for use in a system.

Type of extent – 'D' if data, 'I' if index.

Flags:

- X'80' data RBA with sequence set
- X'40' sequence set RBA with data
- X'20' index replication
- X'10' volume mount flag
- X'08' device contains more than 256 cylinders
- X'04' index for RPS device
- X'02' extent is located on FBA
- X'01' extent is located on an ECKD device

Low extent (CCHH) - the device address (that is, CC = cylinder and HH = track) of the beginning of the extent.

High extent (CCHH) - the device address (that is, CC = cylinder and HH = track) of the end of the extent.

Low RBA - a hexadecimal field containing the RBA (relative byte address) of the beginning of the extent. For an extended-addressed KSDS, this field contains the first CI in the extent.

High RBA - a hexadecimal field containing the RBA (relative byte address) of the end of the extent. For an extended-addressed KSDS, this field contains the last CI in the extent.

Usage

The header is 32 bytes in length. If the value specified for length is large enough, VSE/VSAM returns:

- header;
- physical device characteristics area;
- all rows of Extent Information Area (their number is calculated as “Number of data extents” + “Number of index extents” fields of the header).

If length is not large enough for the whole output (but more than 32 bytes), VSE/VSAM returns as much information as fits, the return code 4, and the reason code 9. Check the field “Total length used (or required) by VSAM” in the header for the required space length, and issue another SHOWCB call specifying LENGTH large enough to contain all the matrix information.

In case user specifies LENGTH less than 32 bytes, VSE/VSAM will reject the request and issue the return code 4 and the reason code 21. Recompile the program with LENGTH bigger than 32 bytes.

Example of an LSR Matrix Call

SHOWCB AREA=USER_AREA, LENGTH=800, SHAREPL=1, FIELDS=(LSRINF)

LSR Input Information

For descriptions of the ACB, RPL, and BLDVRP macros, please see Chapter 12 “VSE/VSAM Macros Descriptions” in the “VSE/VSAM User's Guide and Application Programming”.

LA	R2, BLDVRPA		
BLDVRP	MF=(E, (R2))		
<hr/>			
LSREA	DS OF		
BLDVRPA	BLDVRP KEYLEN=16,		X
	BUFFERS=(8192(20), 512(4)),		X
	STRNO=20,		X
	MF=L,		X
	SHRPOOL=1		
<hr/>			
ACB1	ACB	DDNAME=KSDS, MACRF=(KEY, SEQ, OUT, LSR),	X
		STRNO=9, SHRPOOL=1, BUFND=03, BUFNI=03	
RPL11	RPL	ACB=ACB1, AREA=REC1, AREALEN=40, RECLEN=40, ARG=KEY1,	X
		OPTCD=(KEY, SEQ, NSP, KEQ, MVE)	
ACB2	ACB	DDNAME=KSDS2, MACRF=(KEY, SEQ, OUT, LSR),	X
		STRNO=2, SHRPOOL=1, BUFND=02, BUFNI=02	
RPL21	RPL	ACB=ACB2, AREA=REC2, AREALEN=40, RECLEN=40, ARG=KEY2,	X
		OPTCD=(KEY, SEQ, NSP, KEQ, MVE)	
ACB3	ACB	DDNAME=KSDS3, MACRF=(KEY, SEQ, OUT, LSR),	X
		STRNO=2, SHRPOOL=1, BUFND=02, BUFNI=02	
RPL31	RPL	ACB=ACB3, AREA=REC3, AREALEN=40, RECLEN=1000, ARG=KEY3,	X
		OPTCD=(KEY, SEQ, NSP, KEQ, MVE)	
ACB41	ACB	DDNAME=KSDS4, MACRF=(KEY, SEQ, OUT, LSR),	X
		STRNO=2, SHRPOOL=1, BUFND=03, BUFNI=03	
ACB61	ACB	DDNAME=KSDS6, MACRF=(KEY, SEQ, OUT, LSR),	X
		STRNO=2, SHRPOOL=1, BUFND=03, BUFNI=03	
ACB71	ACB	DDNAME=KSDS7, MACRF=(KEY, SEQ, OUT, LSR),	X
		STRNO=2, SHRPOOL=1, BUFND=03, BUFNI=03	
ACB16	ACB	DDNAME=KSDS16, MACRF=(KEY, SEQ, OUT, LSR),	X
		STRNO=2, SHRPOOL=1, BUFND=03, BUFNI=03	
ACB20	ACB	DDNAME=KSDS20, MACRF=(KEY, SEQ, OUT, LSR),	X
		STRNO=2, SHRPOOL=1, BUFND=03, BUFNI=03	

LSR Matrix Output

```

00000320 00000190 00000010 00000002 00200000 00080024 10 .....
| | | | | | ^=====LENGTH of cluster row=x'24'=36
| | | | | | ^=====NUM of cluster rows=8
| | | | | | ^=====len of buffer row=x'20'=32
| | | | | | ^=====NUM OF buffer ROWS=2
| | | | | | ^=====LEN OF FIXED AREA=x'10'=16
| | | | | | ^=====AREA NEEDED FOR VSAM=x'190'=400
| | | | | | ^=====AREA SUPPLIED BY USER=x'320'=800
-----D.....
00000000 00000000 00010014 00030011 00000000 00000000 0200C400 00000004 .....
| | | | | | | | ^=====NUMBER OF BUFFERS=4
| | | | | | | | ^=====FLAGS=RESERVED=0
| | | | | | | | ^=====TYPE OF BUF='D'
| | | | | | | | ^=====SIZE OF BUFFERS=x'200'=512
| | | | | | | | ^=====RESERVED ( 6 bytes)=0
| | | | | | | | ^=====RESERVED
| | | | | | | | ^=====NUMBER OF FREE STRINGS=x'11'=17
| | | | | | | | ^=====N OF ACTIVE STRINGS=3
| | | | | | | | ^=====TOTAL NUMBER OF STR=x'14'=20
| | | | | | | | ^=====SHR POOL NUMBER=1
| | | | | | | | ^=====RESERVED2=0
| | | | | | | | ^=====RESERVED1=0
-----D.....
00000000 00000004 00000000 00000000 00000000 00000000 2000C400 00000014 .....
| | | | | | | | ^=====NUMBER OF BUFFERS=x'14'=20
| | | | | | | | ^=====FLAGS=RESERVED=0
| | | | | | | | ^=====TYPE OF BUF='D'
| | | | | | | | ^=====SIZE OF BUFFERS=x'2000'=8192
| | | | | | | | ^=====NUM OF NON-USER-INIT WR=0
| | | | | | | | ^=====NUM OF USER-INIT WRITES=0
| | | | | | | | ^=====NUM OF RETR-REQ WITHOUT IO=0
| | | | | | | | ^=====NUM OF BUFFER-READS=0
| | | | | | | | ^=====NUM OF FREE BUFFERS=4
| | | | | | | | ^=====NUM OF MOD BUF=0
-----D.....
00000000 00000008 0000000B 00000015 00000009 00000000 D2E2C4E2 40404040 .....KSDS
| | | | | | | | ^=====DDNAME=KSDS
| | | | | | | | ^=====NUM OF NON-USER-INIT WR=0
| | | | | | | | ^=====NUM OF USER-INIT WRITES=9
| | | | | | | | ^=====NUM OF RETR-REQ WITHOUT IO=x'15'=21
| | | | | | | | ^=====NUM OF BUFFER-READS=x'B'
| | | | | | | | ^=====NUM OF FREE BUFFERS=8
| | | | | | | | ^=====NUM OF MODIFIED BUFFERS=0
-----KSDS
C2000001 00002000 00000007 00002000 00000007 00000000 00000000 D2E2C4E2 B.....
| | | | | | | | ^=====DDNAME=KSDS3
| | | | | | | | ^=====RESERVED2=0
| | | | | | | | ^=====RESERVED1=0
| | | | | | | | ^=====NUM OF INDEX BUFFERS=7
| | | | | | | | ^=====SIZE OF INDEX BUFFERS=x'2000'=8192
| | | | | | | | ^=====NUM OF DATA BUFFERS=7
| | | | | | | | ^=====SIZE OF DATA BUFFERS=x'2000'=8192
| | | | | | | | ^=====NUM OF ACTIVE STRINGS=1
| | | | | | | | ^=====TYPE OF CLUSTER=B
-----B.....
F3404040 C2000001 00002000 00000005 00002000 00000005 00000000 00000000 3 B.....
| | | | | | | | ^=====RESERVED2=0
| | | | | | | | ^=====RESERVED1=0
| | | | | | | | ^=====NUM OF INDEX BUFFERS=5
| | | | | | | | ^=====SIZE OF INDEX BUFFERS=x'2000'=8192
| | | | | | | | ^=====NUM OF DATA BUFFERS=5
| | | | | | | | ^=====SIZE OF DATA BUFFERS=x'2000'=8192
| | | | | | | | ^=====NUM OF ACTIVE STRINGS=1
| | | | | | | | ^=====TYPE OF CLUSTER=B
-----B.....
D2E2C4E2 F2404040 C2000001 00002000 00000003 00002000 00000003 00000000 ..... KSDS2 B.....
00000000 D2E2C4E2 F4404040 C2000000 00002000 00000000 00002000 00000000 ..... KSDS4 B.....
00000000 00000000 D2E2C4E2 F6404040 C2000000 00002000 00000000 00002000 .....KSDS6 B.....
00000000 00000000 00000000 D2E2C4E2 F7404040 C2000000 00002000 00000000 .....KSDS7 B.....
00002000 00000000 00000000 00000000 D2E2C4E2 F1F64040 C2000000 00002000 .....KSDS16 B.....
00000000 00002000 00000000 00000000 00000000 D2E2C4E2 F2F04040 C2000000 .....KSDS20 B.....
    
```

Example of an Extent Matrix Call

SHOWCB AREA=USER_AREA, LENGTH=300, ACB=ACB1, FIELDS=(EXTINF)

Extent Matrix Output

```

0000012C 10
^=====USER'S AREA=X'12C'=300
000000E0 00000060 00000001 00300000 00010030 00000000 00000000 E5E2C5D9 .....VSER
^=====VOLID=VSER02
^=====RESERVED2
^=====RESERVED1
^=====LEN OF INDEX EXT ROW
^=====IND EXTENTS=1
^=====LEN OF DATA EXT ROW
^=====DATA EXTENTS=1
^=====FIXED AREA LEN=X'60'=96
^=====VSAM NEEDS=X'E0'=224
FOF2C426 00000800 0000A800 0009D800 00000001 00000015 0000000F 0000000F 02D.....Q.....
^=====TRACKS PER CYL=X'F'=15
^=====TRACKS PER CA=X'F'=15
^=====PHYS BLOCKS PER TRACK=X'15'=21
^=====PHYS BLOCKS PER CI=X'1'
^=====NUM BYTES PER CA=X'9D800'
^=====NUM BYTES PER TRACK=X'A800'
^=====PHYS BLOCK SIZE=X'800'
^=====FLAGS=X'26'
^=====TYPE OF EXT='D'
0000A800 00000000 00000000 E5E2C5D9 F0F2C926 00000E00 0000B600 0000B600 .....VSER02I.....
^=====NUM BYTES PER CA=X'0000B600'
^=====NUM BYTES PER TRACK=X'0000B600'
^=====PHYS BLOCK SIZE=X'E00'
^=====FLAGS=X'26'
^=====TYPE OF EXT='I'
^=====VOLID=VSER02
^=====RESERVED2
^=====RESERVED1
^=====NUM PHYS BLOCKS PER CA(FBA only)

00000001 0000000D 00000001 0000000F 0000B600 00000000 00000000 E5E2C5D9 .....VSER
^=====VOLSER=VSER02
^=====RESERVED2
^=====RESERVED1
^=====NUM PHYS BLOCKS PER CA(FBA only)
^=====TRACKS PER CYL=X'F'
^=====TRACKS PER CA=X'1'
^=====PHYS BLOCKS PER TRACK=D
^=====PHYS BLOCKS PER CI=X'1'
FOF2C415 00070000 00000000 0009000E 00000000 00000000 00000000 00000000 02D.....
^=====HIGH RBA=X'001D87FF'=1935359
^=====LOW RBA=0
^=====HIGH EXTENT=X'0009000E'
^=====LOW EXTENT=X'00070000'
^=====FLAGS=X'15'
^=====TYPE OF EXT='D'
001D87FF 00000000 00000000 E5E2C5D9 F0F2C915 000A0000 00000000 000A000E ..g.....VSER02I.....
^=====HIGH EXT=X'000A000E'
^=====LOW EXT=X'000A0000'
^=====FLAGS=X'15'
^=====TYPE OF EXT='I'
^=====VOLSER=VSER02
^=====RESERVED2
^=====RESERVED1
00000000 00000000 00000000 00000000 000AA9FF 00000000 00000000 00000000 .....Z.....
^=====RESERVED2
^=====RESERVED1
^=====HIGH RBA=X'000AA9FF'=698879
^=====LOW RBA = 0

```

New Fields Supported by the SHOWCB Macro

Nine new keywords for the FIELDS parameter are supported by SHOWCB in order to enable user to obtain status information for open VSAM datasets.

SHOWCB FIELD	Length	Description
IDACB	4	ACB identifier
IDDOS	4	DOS DTF identifier
CDBUF	4	Count of data buffers
CIBUF	4	Count of index buffers
CNAME	44	Cluster ID
CIPCA	4	Number of control intervals per control area
LNEST	4	Local number of index levels
BFREE	4	Number of unassigned buffers
OPENOBJ	4	AMS Flag byte. With AMS flag you can determine whether the opened object is a path, a base cluster, or an alternate index: X'80'=ALTERNATE INDEX X'40'=ACCESS VIA PATH X'20'=ACCESS VIA BASE CLUSTER

Example of a SHOWCB Call

```

SHOWCB   ACB=ACB1, AREA=AREA1, LENGTH=100, FIELDS=(IDACB, IDDOS, X
          CDBUF, CIBUF, CIPCA, LNEST, BFREE, OPENOBJ, CNAME)
LTR      R15, R15
BNZ      SHOWERR
. . .
AREA1    DS 0F
IDACB    DS F
IDDOS    DS F
CDBUF    DS F
CIBUF    DS F
CIPCA    DS F
LNEST    DS F
BFREE    DS F
OPENOBJ  DS F
CNAME    DS 44CL
    
```


Remarks

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Comments and Questions

Comments or questions on this documentation are welcome. Please send your comments to:
vsesupportvsam@de.ibm.com